R Notebook

### Setting working Directory to Data lOcation

library(data.table)  
library(factoextra)

## Loading required package: ggplot2

## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':  
##   
## between, first, last

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyverse)

## -- Attaching packages ----------------------------------------------------- tidyverse 1.2.1 --

## v tibble 1.4.1 v purrr 0.2.4  
## v tidyr 0.7.2 v stringr 1.2.0  
## v readr 1.1.1 v forcats 0.2.0

## -- Conflicts -------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::between() masks data.table::between()  
## x dplyr::filter() masks stats::filter()  
## x dplyr::first() masks data.table::first()  
## x dplyr::lag() masks stats::lag()  
## x dplyr::last() masks data.table::last()  
## x purrr::transpose() masks data.table::transpose()

train = fread("E:/USA/Projects/Research/R\_code/w6/train\_clust.csv",data.table = T)  
train = train[,-1]  
test = fread("E:/USA/Projects/Research/R\_code/w6/test\_clust.csv",data.table = T)  
test = test[,-1]

Here I have built Custom Function to create Features

features = function(data){  
 newdata = NULL  
 mean\_speed = as.data.frame( rep(0,dim(data)[1]))  
 mean\_acc\_lot =as.data.frame( rep(0,dim(data)[1]))  
 mean\_acc\_lan = as.data.frame(rep(0,dim(data)[1]))  
 sd\_speed = as.data.frame(rep(0,dim(data)[1]))  
 sd\_acc\_lot = as.data.frame(rep(0,dim(data)[1]))  
 sd\_acc\_lat = as.data.frame(rep(0,dim(data)[1]))  
 max\_speed = as.data.frame(rep(0,dim(data)[1]))  
 max\_acc\_lot = as.data.frame(rep(0,dim(data)[1]))  
 max\_acc\_lat = as.data.frame(rep(0,dim(data)[1]))  
 min\_speed = as.data.frame(rep(0,dim(data)[1]))  
 min\_acc\_lot = as.data.frame(rep(0,dim(data)[1]))  
 min\_acc\_lat = as.data.frame(rep(0,dim(data)[1]))  
 for (i in c(1:dim(data)[1])) {  
 mean\_speed[i,] = mean(unlist(data[i,4:64]))  
 mean\_acc\_lot[i,] = mean(unlist(data[i , 65:125]))  
 mean\_acc\_lan[i,] = mean(unlist(data[i, 126:186]))  
 sd\_speed[i,] = sd((unlist(data[ i,4:64])))  
 sd\_acc\_lot[i,] = sd((unlist(data[i , 65:125])))  
 sd\_acc\_lat[i,] = sd((unlist(data[i , 126:186])))  
 max\_speed[i,] = max((unlist(data[ i,4:64])))  
 max\_acc\_lot[i,] = max((unlist(data[i , 65:125])))  
 max\_acc\_lat[i,] = max((unlist(data[i , 126:186])))  
 min\_speed[i,] = min((unlist(data[ i,4:64])))  
 min\_acc\_lot[i,] = min((unlist(data[i , 65:125])))  
 min\_acc\_lat[i,] = min((unlist(data[i , 126:186])))  
 }  
 newdata =as.data.table(cbind(mean\_speed,mean\_acc\_lot,mean\_acc\_lan, sd\_speed,sd\_acc\_lot,sd\_acc\_lat,  
 max\_speed,max\_acc\_lot,max\_acc\_lat,min\_speed,mean\_acc\_lot,mean\_acc\_lan))  
 colnames(newdata) = c("mean\_speed","mean\_acc\_lot","mean\_acc\_lan", "sd\_speed","sd\_acc\_lot","sd\_acc\_lat","max\_speed",  
 "max\_acc\_lot","max\_acc\_lat","min\_speed", "mean\_acc\_lot","mean\_acc\_lan")  
 return(newdata)  
}

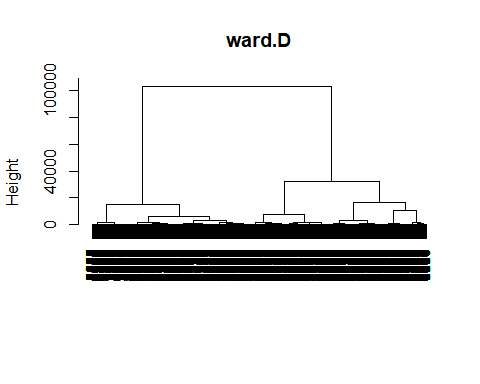
### Creating Data

train\_feat = features(train)  
test\_feat = features(test)

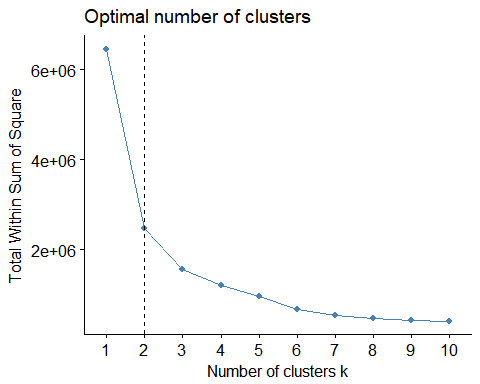
hc\_ward=hclust(dist(train\_feat), method="ward.D")

### Questions related to type of Dissimilarity measure to use?

plot(hc\_ward,main="ward.D", xlab="", sub="", cex=.9)

 Here we can see only 2 clusters.

fviz\_nbclust(train\_feat, hcut, method = "wss",hc\_method = "ward.D", main = "Ward.D") +  
 geom\_vline(xintercept = 2, linetype = 2)

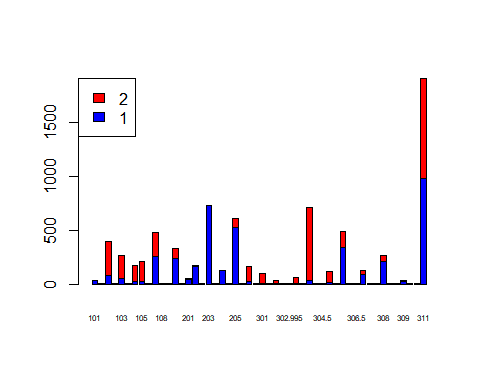


hc\_ward\_cut = cutree(hc\_ward,k=2)  
hc\_ward\_cut2 = cutree(hc\_ward,k=3)

index = fread("E:/USA/Projects/Research/R\_code/w6/index\_all.csv")  
index$new = paste(index$RunName,index$win60s)  
train$new = paste(train$RunName,train$win60s)  
  
index\_train = as.data.table(train$new)  
colnames(index\_train)='new'  
index\_train = merge(index\_train, index[,c('c','new')],by = 'new')  
  
index\_train1 = cbind(index\_train,as.data.table(hc\_ward\_cut))

index\_train1$c = as.factor(index\_train1$c)  
index\_train1$hc\_ward\_cut =as.factor(index\_train1$hc\_ward\_cut)

ind = group\_by(index\_train1[,2:3], c) %>% summarize(size = length(hc\_ward\_cut), frq1 = summary(hc\_ward\_cut)[1],frq2 = summary(hc\_ward\_cut)[2])  
  
barplot(height = t(ind[,c(3,4)]), names.arg = ind$c,col=c("blue","red"),legend.text = c("1","2"),args.legend = list(x = "topleft"),axisnames = T,cex.names = 0.5)



index\_train2 = cbind(index\_train,as.data.table(hc\_ward\_cut2))  
index\_train2$c = as.factor(index\_train2$c)  
index\_train2$hc\_ward\_cut2 =as.factor(index\_train2$hc\_ward\_cut2)

ind1 = group\_by(index\_train2[,2:3], c) %>% summarize(size = length(hc\_ward\_cut2), frq1 = summary(hc\_ward\_cut2)[1],frq2 = summary(hc\_ward\_cut2)[2],frq3 = summary(hc\_ward\_cut2)[3])  
  
barplot(height = t(ind1[,c(3,4,5)]), names.arg = ind1$c,col=c("blue","red","green"),legend.text = c("1","2","3"),args.legend = list(x = "topleft"),axisnames = T,cex.names = 0.5)

